

**Perancangan Wall Follower Berbasis Proportional Dan Derivative
pada robot pemadam api berkaki 6 “DOME INA” Pada Trinity
Collage Fire fighting Home Robot Contest (TCFFHRC) 2019**

SKRIPSI



Disusun Oleh :

Ken Dedes Maria Kunthy

NIM : 201510130311179

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH MALANG

2019

LEMBAR PERSETUJUAN

**Perancangan Wall Follower Berbasis Proportional dan Derivative Pada Robot Pemadam
Api Berkaki 6 “DOME INA” Pada *Trinity Collage Fire Fighting Home Robot Contest*
(TCFFHRC) 2019**

Diajukan Sebagai Salah Satu Syarat Memperoleh Gelar Sarjana (S1)
Teknik Elektro Universitas Muhammadiyah Malang

Disusun Oleh:

Ken Dedes Maria Kunthy

201510130311179

Diperiksa dan disetujui oleh:

Pembimbing I



Ir. Nur Alif Mardiyah, MT

NIP.UMM: 10892030257

Pembimbing II



Machmud Effendy, ST., M.Eng

NIP.UMM: 10802030363

LEMBAR PENGESAHAN

**Perancangan Wall Follower Berbasis Proportional Dan Derivative Pada Robot
Pemadam Api Berkaki 6 “DOME INA” Pada Trinity Collage Fire Fighting Home Robot
Contest (TCFFHRC) 2019**

Skripsi ini Disusun Sebagai Salah Satu Syarat Memperoleh Gelar Sarjana (S1) Teknik
Elektro Universitas Muhammadiyah Malang

Oleh:

Ken Dedes Maria Kunthy

201510130311179

Tanggal Ujian : 18 Maret 2019

Tanggal Wisuda : 27 April 2019

Disetujui Oleh :

1. Ir. Nur Alif Mardiyah, MT
NIP.UMM: 10892030257

(Pembimbing 1)

2. Machmud Effendy, ST, M.Engl
NIP.UMM: 10802030363

(Pembimbing II)

3. Widianto, ST., MT
NIP.UMM: 10817090612

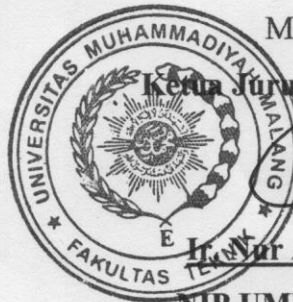
(Penguji I)

4. M. Chasran Hasani, ST, MT
NIP.UMM: 196808071995031003

(Penguji II)

Mengetahui,

Ketua Jurusan Teknik Elektro



Ir. Nur Alif Mardiyah, MT

NIP.UMM: 10892030257

LEMBAR PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama : **Ken Dedes Maria Kunthy**
Tempat/Tgl. Lahir : **Malang, 11 Desember 1997**
NIM : **201510130311179**
Fakultas/Jurusan : **TEKNIK/TEKNIK ELEKTRO**


Dengan ini saya menyatakan bahwa Skripsi dengan judul "**Perancangan Wall Follower Berbasis Proportional Dan Derivative Pada Robot Pemadam Api Berkaki 6 "DOME INA" Pada Trinity Collage Fire fighting Home Robot Contest (TCFFHRC) 2019**" beserta seluruh isinya adalah karya saya sendiri dan bukan merupakan karya tulis orang lain, baik sebagian maupun seluruhnya, kecuali dalam bentuk kutipan yang telah disebutkan sumbernya.

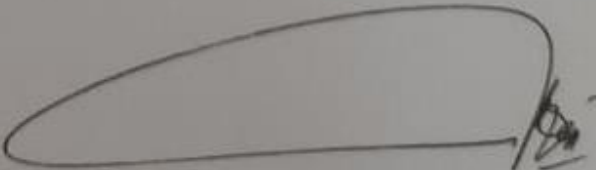
Demikian surat pernyataan ini saya buat dengan sebenar-benarnya. Apabila kemudian ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya ini, atau ada klaim dari pihak lain terhadap keaslian karya saya ini maka saya siap menanggung segala bentuk resiko/sanksi yang berlaku.

Malang, 11 Juli 2019

METERAI
TEMPEL
0A28EAF964130240
6000
ENAM RIBURUPIAH
Ken Dedes Maria Kunthy

Mangetahui,


Ir. Nur Alif Mardiyah, MT
NIP.UMM: 10892030257


Machmud Effendy, ST., M.Eng
NIP.UMM: 10802030363

KATA PENGANTAR

Puji dan syukur senantiasa kita panjatkan ke hadirat Allah SWT atas segala nikmat, kekuatan, taufik serta hidayah-Nya. Shalawat serta salam semoga tercurah kepada Rasulullah SAW, keluarga sahabat dan para pengikut setianya, Amin. Atas kehendak Allah sajalah, penulis dapat menyelesaikan proyek akhir yang berjudul :

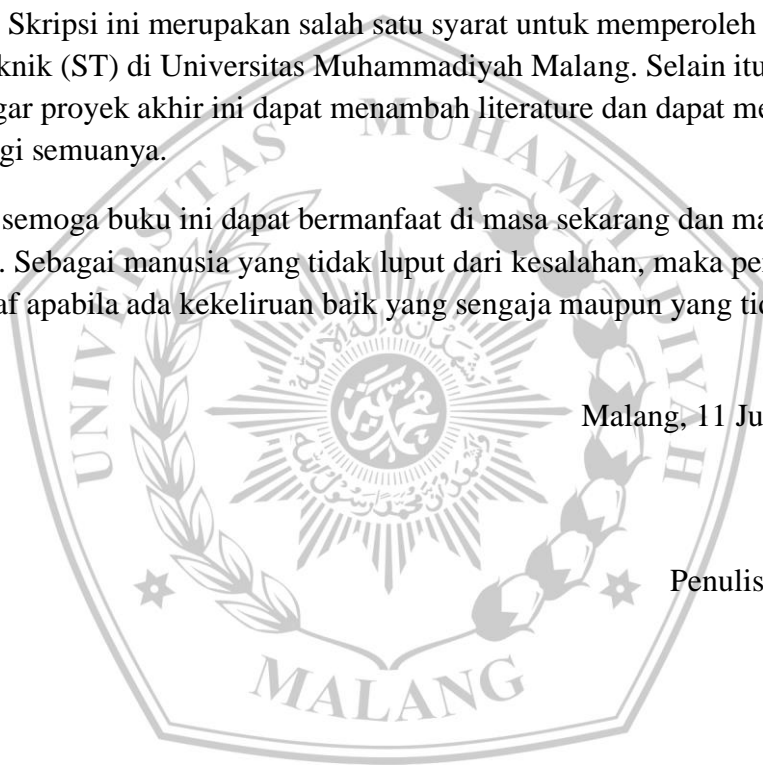
“Perancangan Wall Follower Berbasis Proportional Dan Derivative Pada Robot Pemadam Api Berkaki 6 “DOME INA” Pada *Trinity Collage Fire Fighting Home Robot Contest (TCFFHRC) 2019*”

Pembuatan Skripsi ini merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik (ST) di Universitas Muhammadiyah Malang. Selain itu penulis berharap agar proyek akhir ini dapat menambah literature dan dapat memberikan manfaat bagi semuanya.

Akhir kata semoga buku ini dapat bermanfaat di masa sekarang dan masa mendatang. Sebagai manusia yang tidak luput dari kesalahan, maka penulis mohon maaf apabila ada kekeliruan baik yang sengaja maupun yang tidak sengaja.

Malang, 11 Juli 2019

Penulis



DAFTAR ISI

i. JUDUL	i
ii. LEMBAR PERSETUJUAN	ii
iii. LEMBAR PENGESAHAN	iii
iv. LEMBAR PERNYATAAN	iv
v. ABSTRAK	v
vi. KATA PENGANTAR	vii
vii. LEMBAR PERSEMBAHAN	viii
viii. DAFTAR ISI	ix
ix. DAFTAR GAMBAR	xii
x. DAFTAR TABEL	xiv
1 BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat	3
1.6 Sistematika Penulisan	3
2 BAB II DASAR TEORI	5

2.1	Piranti Masukan	5
2.1.1	Sensor Infrared (Sharp GP2Y0A41)	5
2.1.2	Sensor Ultrasonik (PING)	6
2.2	Piranti Keluaran	7
2.2.1	Motor Servo AX-18A	7
2.3	STM32F407	7
2.4	PD (Proportional dan Derivatif)	8
2.4.2	Kontroller P	9
2.4.1	Kontroller PD (Proportional Derivatif)	9
2.5	Wall Follower	9
3	BAB III PERANCANGAN DAN PEMBUATAN SISTEM	11
3.1	Perancangan dan Pembuatan Mekanik Robot	11
3.1.1	Posisi Peletakan Sensor	11
3.2	Perancangan Perangkat Keras	13
3.3	Perancangan Instrumentasi	13
3.3.1	PING	13
3.3.2	Sharp	14
3.4	Wall Follower	15
3.5	PD Pada Wall Follower	17
4	BAB IV PENGUJIAN DAN ANALISIS	19

4.1	Pengujian Sensor Infrared Sharp	19
4.4.1	Tujuan	19
4.1.2	Peralatan.....	19
4.1.3	Blok Diagram.....	19
4.1.4	Persiapan	20
4.1.5	Hasil Dan Analisis	20
4.2	Pengujian PD Pada Wall Follower	21
4.2.1	Tahapan Pengujian.....	22
4.2.2	Pengujian Sistem.....	22
4.2.2.1	Pengujian Sistem Kontrol Proportional (P)	22
4.2.2.2	Pengujian Sistem Kontrol Proportional,Derivatif (PD).....	24
4.3	Pengujian PD Pada Wall Follower Dengan Furniture Dan Boneka	28
4.3.1	Tahapan Pengujian.....	28
4.3.2	Hasil Dan Analisis	28
5	BAB V PENUTUP	32
5.1	Kesimpulan	32
5.2	Saran	32
	DAFTAR PUSTAKA	33

DAFTAR GAMBAR

Gambar 2.1 Infrared Sensor Sharp GP2Y0A41	5
Gambar 2.2 Sensor Ultrasonik (PING)	6
Gambar 2.3 Sensor Gyroscope Dan Accelerometer MPU6050	6
Gambar 2.4 Motor Servo AX-18	7
Gambar 2.5 STM32F047	8
Gambar 2.6 Wall Follower	10
Gambar 3.1 Mekanik Robot Keseluruhan	11
Gambar 3.2 Mekanik Sensor	12
Gambar 3.3 Perancangan Perangkat Keras	13
Gambar 3.4 Rangkaian Sensor Infrared Sharp	14
Gambar 3.5 Metode Wall Follower	15
Gambar 3.6 Flowchart Wall Follower	16
Gambar 3.7 Blok diagram PD pada Wall Follower	17
Gambar 3.8 Present Value PD Wall Follower	18
Gambar 4.1 Blok Diagram Pengujian Sensor Infrared Sharp	19
Gambar 4.2 Perbandingan Tegangan Dan Jarak	21
Gambar 4.3 Respon Sistem Dengan Kontrol P Pada Kondisi 1	22
Gambar 4.4 Respon Sistem Dengan Kontrol P Pada Kondisi 2	23

Gambar 4.5 Respon Sistem Dengan Kontrol P Pada Kondisi 3	24
Gambar 4.6 Respon Sistem Dengan Kontrol PD Pada Kondisi Ke 1	25
Gambar 4.7 Respon Sistem Dengan Kontrol PD Pada Kondisi Ke 2	26
Gambar 4.8 Respon Sistem Dengan Kontrol PD Pada Kondisi Ke 3	27
Gambar 4.9 Track Dengan Boneka Pada Posisi A	29
Gambar 4.10 Track Dengan Boneka Pada Posisi B	29
Gambar 4.11 Track Dengan Boneka Pada Posisi C	30



DAFTAR TABEL

Tabel 2.1 Pengaruh Proportional Dan Derivatif Keadaan Loop Tertutup, Waktu Naik, Overshoot, Saat Turun, Kesalahan Tunak	9
Tabel 4.1 Data Pengujian Sensor Infrared Sharp	20
Tabel 4.2 Jumlah Benturan Saat Menyusuri Keseluruhan Arena	30



Lampiran

Listing Program Ping Sensor

```
#include "mylib_ping.h"

#include "mylib_buzzer.h"

//Ping

#define DELAY_BETWEEN_PING    23

#define TIMEOUT_PING          23

#define IS_PING_NUMBER(PINGN) ((PINGN) < (uint8_t)10)

#define PING1_GPIO_PIN                GPIO_PIN_1
    //LeftFront

#define PING1_GPIO_PORT                GPIOD

#define PING1_GPIO_CLK_ENABLE()        __GPIOD_CLK_ENABLE()

#define PING1_EXTI_IRQn                EXTI1_IRQn

#define PING2_GPIO_PIN                GPIO_PIN_0
    //LeftMiddle

#define PING2_GPIO_PORT                GPIOD

#define PING2_GPIO_CLK_ENABLE()        __GPIOD_CLK_ENABLE()

#define PING2_EXTI_IRQn                EXTI0_IRQn

#define PING3_GPIO_PIN                GPIO_PIN_7
    //LeftRear

#define PING3_GPIO_PORT                GPIOC

#define PING3_GPIO_CLK_ENABLE()        __GPIOC_CLK_ENABLE()

#define PING3_EXTI_IRQn                EXTI9_5_IRQn

#define PING4_GPIO_PIN                GPIO_PIN_6    //Rear

#define PING4_GPIO_PORT                GPIOC

#define PING4_GPIO_CLK_ENABLE()        __GPIOC_CLK_ENABLE()

#define PING4_EXTI_IRQn                EXTI9_5_IRQn

#define PING5_GPIO_PIN                GPIO_PIN_14
    //RightRear

#define PING5_GPIO_PORT                GPIOD

#define PING5_GPIO_CLK_ENABLE()        __GPIOD_CLK_ENABLE()
```



```

#define PING12_GPIO_PORT                                GPIOD

#define PING12_GPIO_CLK_ENABLE()    __GPIOD_CLK_ENABLE()

#define PING12_EXTI_IRQn                EXTI9_5_IRQn

#define PINGx_GPIO_CLK_ENABLE(__INDEX__) do{if((__INDEX__) == 0)
PING1_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 1) PING2_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 2) PING3_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 3) PING4_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 4) PING5_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 5) PING6_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 6) PING7_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 7) PING8_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 8) PING9_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 9) PING8_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 10) PING9_GPIO_CLK_ENABLE(); else \

                                if((__INDEX__) == 11) PING10_GPIO_CLK_ENABLE();\

                                }while(0)

uint16_t SumPingError[PINGn];

bool EnableTrigerPing = true;

uint8_t CursorPing = 0;

uint32_t LastTimeGetPing = 0, StartTrigerTime = 0;

uint32_t RisingTimePing, FallingTimePing;

bool EnablePing[PINGn];

__IO float DataPing[PINGn];

bool errorPing[PINGn];

GPIO_TypeDef* PING_PORT[PINGn] = {PING1_GPIO_PORT,

```


PING2_GPIO_PORT,

PING3_GPIO_PORT,

PING4_GPIO_PORT,

PING5_GPIO_PORT,

PING6_GPIO_PORT,

PING7_GPIO_PORT,

PING8_GPIO_PORT,

PING9_GPIO_PORT,

PING10_GPIO_PORT,

PING11_GPIO_PORT,

PING12_GPIO_PORT,};

const uint16_t PING_PIN[PINGn] = {PING1_GPIO_PIN,

PING2_GPIO_PIN,

PING3_GPIO_PIN,

PING4_GPIO_PIN,

PING5_GPIO_PIN,

PING6_GPIO_PIN,

PING7_GPIO_PIN,

PING8_GPIO_PIN,



```

PING9_GPIO_PIN,

PING10_GPIO_PIN,

PING11_GPIO_PIN,

PING12_GPIO_PIN,};

const uint8_t PING_IRQn[PINGn] = {PING1_EXTI_IRQn,

PING2_EXTI_IRQn,

PING3_EXTI_IRQn,

PING4_EXTI_IRQn,

PING5_EXTI_IRQn,

PING6_EXTI_IRQn,

PING7_EXTI_IRQn,

PING8_EXTI_IRQn,

PING9_EXTI_IRQn,

PING10_EXTI_IRQn,

PING11_EXTI_IRQn,

PING12_EXTI_IRQn,};

```

```

void PING_MODE_EXTI_Init(uint8_t PING);

void PING_MODE_GPIO_OUT_Init(uint8_t PING);

void PING_DeInit(uint8_t PING);

```

```

/*****

```

PING FUNCTIONS

```

/
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)

{
    if(RisingTimePing == 0 ){RisingTimePing = micros();}

    else{FallingTimePing = micros();}

}

void GetPing(void)

{
    if(EnableTrigerPing && ((HAL_GetTick() - LastTimeGetPing) >= DELAY_BETWEEN_PING))
    {
        //Bip(500);
        EnableTrigerPing = false;
        PING_MODE_GPIO_OUT_Init(CursorPing);
        HAL_GPIO_WritePin(PING_PORT[CursorPing], PING_PIN[CursorPing],
GPIO_PIN_RESET);
        delayMicroseconds(2);
        HAL_GPIO_WritePin(PING_PORT[CursorPing], PING_PIN[CursorPing],
GPIO_PIN_SET);
        delayMicroseconds(5);
        HAL_GPIO_WritePin(PING_PORT[CursorPing], PING_PIN[CursorPing],
GPIO_PIN_RESET);
        delayMicroseconds(2);

        PING_DeInit(CursorPing);

        StartTrigerTime = HAL_GetTick();

        PING_MODE_EXTI_Init(CursorPing);

        RisingTimePing = 0;

        FallingTimePing = 0;

    }

    else if(((RisingTimePing != 0) && (FallingTimePing != 0) && (!EnableTrigerPing)) ||
(((HAL_GetTick() - StartTrigerTime) >= TIMEOUT_PING) && (!EnableTrigerPing)))

```

```

    {
        if (FallingTimePing != 0)
        {
            DataPing[CursorPing] = (float)((float)(FallingTimePing -
RisingTimePing)/29.034f)/2;

            errorPing[CursorPing] = false;

            LastTimeGetPing = HAL_GetTick();
        }
        else
        {
            BipNoCheck(200);
            errorPing[CursorPing] = true;
            SumPingError[CursorPing]++;
        }
        PING_DeInit(CursorPing);
        do{
            CursorPing++;
            if(CursorPing >= PINGn){CursorPing = 0;}
        }
        while(!EnablePing[CursorPing]);
        EnableTrigerPing = true;
        RisingTimePing = 0;
        FallingTimePing = 0;
    }
}

void PING_MODE_EXTI_Init(uint8_t PING)
{
    GPIO_InitTypeDef GPIO_InitStruct;

    //assert_param(IS_PING_NUMBER(PING));

    /* Enable the PING Clock */

```

```

PINGx_GPIO_CLK_ENABLE(PING);

    /* Configure Button pin as input with External interrupt */

    GPIO_InitStruct.Pin = PING_PIN[PING];

    GPIO_InitStruct.Pull = GPIO_NOPULL;

    GPIO_InitStruct.Mode = GPIO_MODE_IT_RISING_FALLING;

    HAL_GPIO_Init(PING_PORT[PING], &GPIO_InitStruct);

    /* Enable and set Ping EXTI Interrupt to the lowest priority */

    HAL_NVIC_SetPriority((IRQn_Type)(PING_IRQn[PING]), 0, 0);

    HAL_NVIC_EnableIRQ((IRQn_Type)(PING_IRQn[PING]));

}

void PING_MODE_GPIO_OUT_Init(uint8_t PING)
{
    GPIO_InitTypeDef GPIO_InitStruct;

    //assert_param(IS_PING_NUMBER(PING));

    /* Enable the BUTTON Clock */
    PINGx_GPIO_CLK_ENABLE(PING)

    /* Configure Button pin as input */

    GPIO_InitStruct.Pin = PING_PIN[PING];

    GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;

    GPIO_InitStruct.Pull = GPIO_NOPULL;

    GPIO_InitStruct.Speed = GPIO_SPEED_FAST;

    HAL_GPIO_Init(PING_PORT[PING], &GPIO_InitStruct);

}

void PING_DeInit(uint8_t PING)
{
    //assert_param(IS_PING_NUMBER(PING));

    HAL_GPIO_DeInit(PING_PORT[PING], PING_PIN[PING]);

    HAL_NVIC_DisableIRQ((IRQn_Type)(PING_IRQn[PING]));

}

```

```

void AllPingEnable(void)

{

    for(int i=0;i<PINGn;i++){EnablePing[i] = true;}

}

void AllPingDisable(void)

{

    for(int i=0;i<PINGn;i++){EnablePing[i] = false;}

}

void EXTI0_IRQHandler(void)

{

    HAL_GPIO_EXTI_IRQHandler(PING2_GPIO_PIN);

}

void EXTI1_IRQHandler(void)

{

    HAL_GPIO_EXTI_IRQHandler(PING1_GPIO_PIN);

}

void EXTI9_5_IRQHandler(void)

{

    HAL_GPIO_EXTI_IRQHandler(PING3_GPIO_PIN);//

    HAL_GPIO_EXTI_IRQHandler(PING4_GPIO_PIN);//

    HAL_GPIO_EXTI_IRQHandler(PING11_GPIO_PIN);//

    HAL_GPIO_EXTI_IRQHandler(PING12_GPIO_PIN);//

}

void EXTI15_10_IRQHandler(void)

{

    HAL_GPIO_EXTI_IRQHandler(PING5_GPIO_PIN);//

    HAL_GPIO_EXTI_IRQHandler(PING6_GPIO_PIN);//

    HAL_GPIO_EXTI_IRQHandler(PING7_GPIO_PIN);//

    HAL_GPIO_EXTI_IRQHandler(PING8_GPIO_PIN);//

```



```

        HAL_GPIO_EXTI_IRQHandler(PING9_GPIO_PIN);//

        HAL_GPIO_EXTI_IRQHandler(PING10_GPIO_PIN);//

    }

```

Listing Program Sharp IR Sensor

```

#include "mylib_sharp.h"

#include "math.h"

#include "mylib_adc.h"

#include "mylib_buzzer.h"

/*-----
SHARP CONFIGURATION
-----*/

#define SHARP1_SENSOR ADC_CHANNEL_3
#define SHARP2_SENSOR ADC_CHANNEL_1
#define SHARP3_SENSOR ADC_CHANNEL_2
#define SHARP4_SENSOR ADC_CHANNEL_0

//Sharp

float distanceIR[SharpIRn];

const uint32_t SharpADCChanel[SharpIRn] =
{SHARP1_SENSOR,SHARP2_SENSOR,SHARP3_SENSOR,SHARP4_SENSOR,};

void sharpInit(void)

{

    for(char i=0;i<SharpIRn;i++)

    {

        ADCInit(SharpADCChanel[i]);

    }

}

void getSharp(void)

{

```

```

float analogIR[SharpIRn];

for(char i=0;i<SharpIRn;i++)
{
    analogIR[i]=analogRead(SharpADCC Chanel[i]);

    analogIR[i]=analogRead(SharpADCC Chanel[i]);
}

for(char i=0;i<SharpIRn;i++)
{
    if(analogIR[i] != 0)
    {
        analogIR[i] = (float)analogIR[i]/4095;
        analogIR[i] = (float)analogIR[i]*3.3f;
        analogIR[i] = 27.86f*pow(analogIR[i],-1.15f);//10-80cm
        distanceIR[i] = (distanceIR[i]*0.6f)+(analogIR[i]*0.4f);
    }
    else{ while(1){ BipError1();}}
}
//
}

```



DAFTAR PUSTAKA

- [1] Hidayat, K. 2012. Perancangan Robot Hexapod Pemadam Api pada Kontes Robot Cerdas Indonesia 2011 (Skripsi). Malang: Universitas Muhammadiyah Malang
- [2] T. College, "Trinity College Fire-Fighting Home Robot Contest 2019 Rules V1.9", Trinity College, 2019.
- [3] Fikri M. K., dkk. "Perancangan dan Implementasi *Real Seagway* pada *Sketboard* Roda Satu Menggunakan *Gyroscope* dan *Accelerometer*". J-PTIHK. Vol. 1, No. 1. Jan. 2017.
- [4] *STM32407 Datasheet*, www.st.com/en/microcontrollers/stm32f407-417.html
- [5] Zuhal, Zhanggishan. 2004. Prinsip Dasar Elektro Teknik. Jakarta: Gramedia Pustaka Utama
- [6] Robotis, "User's Manual Dynamixel AX-18", Robotis Co.,Ltd.
- [7] O. Sorin dan N. Mircea, "The Modeling of the Hexapod Mobile Robot Leg and Associated Interpolated Movements While Stepping," *System Theory, Control and Computing (ICSTCC)*, 2012 16th International Conference on, 2012.
- [8] Braunl , Thomas. 2003. *Embedded Robotics: Mobile Robot Design and Application with Embedded System*. Perth: Springer
- [9] Lin J. N, Song S. M. *Modeling gait transition of quadrupeds and their generalization with CMAC neural networks*. *IEEE Transactions on system, man, and cybetics, Part C*, 2002, 32(3) 177-189.



UNIVERSITAS MUHAMMADIYAH MALANG
FAKULTAS TEKNIK
JURUSAN TEKNIK ELEKTRO & D3 TEKNIK ELEKTRONIKA
Jl. Raya Tlogomas 246 Malang 65144 Telp. 0341 - 464318 Ext. 129, Fax. 0341 - 460782

FORM CEK PLAGIASI LAPORAN TUGAS AKHIR

Nama Mahasiswa : Ken Dedes Maria Kunthy
NIM : 201510130311179
Judul TA : Perancangan Wall Follower Berbasis Proportional Dan Derivative Pada Robot Pemadam Api Berkaki 6 DOME INA Pada Trinity Fire Fighting Home Robot Contest (TFFHRC) 2019

Hasil Cek Plagiarisme dengan Turnitin

No.	Komponen Pengecekan	Nilai Maksimal Plagiasi (%)	Hasil Cek Plagiasi (%) *
1.	Bab 1 – Pendahuluan	10 %	5%
2.	Bab 2 – Studi Pustaka	25 %	11%
3.	Bab 3 – Metodologi Penelitian	35 %	33%
4.	Bab 4 – Pengujian dan Analisis	15 %	7%
5.	Bab 5 – Kesimpulan dan Saran	5 %	0%
6.	Publikasi Tugas Akhir	20 %	16%

Mengetahui,

Dosen Pembimbing I,

(Ir. Nur Alif Mardiyah, MT)
NIP.UMM: 10892030257

Dosen Pembimbing II,

(Machmud Effendy, ST., M.Eng)
NIP.UMM: 10802030363